Journal of Bionic Engineering, Volume 5, Issue 4, December 2008, Pages 299-307,
ISSN 1672-6529, DOI: 10.1016/S1672-6529(08)60174-2

# An efficient multi-scale modelling approach for ssDNA motion in fluid flow

M. Benke, E. Shapiro and D. Drikakis

Fluid Mechanics & Computational Science Group

Aerospace Sciences Department, Cranfield University

Cranfield, Bedfordshire, MK43 0AL, UK

Tel.: +44 (0) 1234 754749

Email: d.drikakis@cranfield.ac.uk

Abstract

The paper presents a multi-scale modelling approach for simulating macromolecules in fluid flows. The focus is mainly on macromolecule transport at low number densities which is frequently encountered in biomedical devices such as separators, and detection and analysis systems. Modelling the transport process is a challenging problem due to the wide range of physical scales involved. On one hand, the continuum approach is not valid for low solute concentrations and, on the other hand, the large time scales of the fluid flow make molecular simulations prohibitively expensive. The proposed multi-scale modelling strategy enables simulation of individual macromolecules on macroscopic time scales. The particular class of multi-scale models considered in this paper is the meta-models. Meta-models are based on the coupled solution of fluid flow equations and equations of motion for a simplified mechanical model of macromolecules. Meta-models often rely on particle-corrector algorithms, which impose length constraints on the mechanical model. Lack of robustness of the particle-corrector algorithm employed can lead to slow convergence and numerical instability. A new FAst Linear COrrector (FALCO) algorithm is introduced in this paper, which significantly improves computational efficiency in comparison with the widely used SHAKE algorithm. Validation of the new particle corrector against a simple analytic solution is performed and improved convergence is demonstrated for ssDNA motion in a lid-driven micro-cavity.

Keywords: multi-scale modelling, DNA, macromolecule transport, meta-modelling, particle corrector.

## 1 Introduction

In the past decade micro- and nanofluidic devices have emerged as revolutionary tools for a wide range of applications in chemical engineering and bioengineering [1-6]. Microfluidic devices are extensively used to enhance mixing [7,8] and separation [9,10], drug delivery [11,12] and bioanalysis systems [13,14]. The transport of macromolecules in microfluidic devices is relevant to a number of biomedical applications including separation and detection [5,6,15-18]. Numerical modelling of macromolecule transport enables prediction and optimisation of microfluidic device performance for a particular application [19-22]. Moreover, simulation results can be directly utilised to guide functionality-oriented design of microfluidics. The key challenge in the numerical modelling of macromolecule transport in microfluidics lies in the inherently multiscale nature of the process, which is governed by a combination of stochastic molecular motion and deterministic macroscopic flow. Different models have been developed in the past to describe transport phenomena occurring on a variety of length scales [23-25]. Pure molecular models [24] provide a detailed description of nanoscale events, however, the computational requirements associated with the molecular modelling [24] would prevent in most cases the direct simulation of the interaction of macromolecules with a continuum scale microflow. On the other side of the scale, macroscale continuum

models based on continuum fluid flow equations and additional advection-diffusion equation for the molecules concentration field can be utilised [26]. However, many applications, in particular bioanalytical and drug delivery systems, feature extremely low solute concentrations [13-14] for which continuum concentration field models are not suitable. An indicative example can be derived from experimental studies of DNA transport in microchannels [18] which indicate that individual DNA strands in a pressure driven microflow tend to migrate to the middle of the microchannel. Contrary to the experimental observations, the application of a continuum level diffusion model would result in spreading the DNA in the microfluidic channel.

A viable alternative to purely molecular or continuum level description is provided by meta-models which have been applied successfully to polymer science[27].

Meta-models couple the continuum level flow equations with a mechanical analogue of the macromolecule. The input from molecular models is required for defining the parameters of the mechanical structure representing the macromolecule of interest [28]. The motion of a macromolecule is then represented by the motion of the corresponding mechanical structure in the force field imposed by the flow under a set of geometrical constraints. Efficient and stable numerical implementation of the particle position corrector algorithm, which is applied to impose the geometrical constraints on the moving mechanical structure, is a key to the overall computational efficiency of the meta-model. Trebotich et al. [28] used the SHAKE algorithm to simulate DNA, however convergence problems were observed when an initial guess for the SHAKE algorithm was far from the final corrected particle positions.

In this paper, a new FAst Linear COrrector (FALCO) algorithm is proposed. The algorithm is derived from purely geometrical considerations and is based on analogue of the Hooke's law [30] with a variable spring constant. The proposed particle corrector is validated using the rotating dumbbell problem [31] for which an analytic solution is available and its performance is demonstrated for a coupled motion of an ssDNA segment in a lid-driven micro-cavity flow.

## 2 Description of models

Experimental observations indicate that ssDNA is a relatively flexible biopolymer [32,33]. Therefore, the mechanical behaviour of ssDNA is predominantly governed by the stretching forces and an appropriate mechanical representation can be constructed using the bead-rod model [27], which has been successfully applied to DNA simulations previously [28]. The model is based on the mechanical representation of the macromolecule as a series of $\mathrm{N}$ beads connected with rigid rods (Figure 1). The motion of the macromolecule in the fluid flow is then described by the set of Newton equations of motion for each individual bead position, $\boldsymbol{r}_i$ given by

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \boldsymbol{F}_i + \boldsymbol{G}_i, \qquad [1]$$

where $m_i$ is the mass of the bead; $\boldsymbol{F}_i$ represents the force exerted on the bead by the flow and $\boldsymbol{G}_i$ is the constraint force which arises from the bond deviations defined as

$$\sigma_{ij} = (\boldsymbol{r}_i - \boldsymbol{r}_j)^2 - d_{ij}^2 \qquad i, j = 1...N, \quad i \neq j, \tag{2}$$

where $d_{ij}$ is the prescribed bond length and $|i - j| = 1$. When the length of the *ij*th bond $|\boldsymbol{r}_i - \boldsymbol{r}_j| = l$ is equal to the prescribed value, there is no constraint force acting on the *i*th and *j*th beads due to this bond.

The equations of motion for the beads have to be solved in conjunction with the incompressible Navier-Stokes system [23] comprising mass

$$\nabla \cdot \boldsymbol{v} = 0 \tag{3}$$

and momentum

$$\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v} \cdot \nabla)\boldsymbol{v} = -\frac{1}{\rho}\nabla p + \upsilon \Delta \boldsymbol{v} + \frac{1}{\rho}\boldsymbol{F}_b \tag{4}$$

conservation equations, where $p$, $\boldsymbol{v}$, $\upsilon$ and $\rho$ denote pressure, fluid velocity, kinematic viscosity and density of fluid respectively. $\boldsymbol{F}_b$ denotes the external forces which are exerted on the fluid flow by the macromolecule. These external forces are generally small in comparison with the convective and viscous terms. Considering the current setup, the ratio of convective and external terms is approximately $10^4$ and the ratio of viscous and external terms is around $10^7$.

Several models are available for the construction of the total constraint forces vector $\boldsymbol{G}_i$. Assuming that each constraint force component acts along the rod connecting corresponding beads and the magnitude of the force is proportional to the difference between the prescribed and actual bond lengths, it is possible to write the general constraint force (e.g. [27]) acting on the *i*th bead as

$$\boldsymbol{G}_i = -\sum_{j=1\, j\neq i}^{N} \lambda_{ij} \frac{\partial \sigma_{ij}}{\partial \boldsymbol{r}_i}, \qquad [5]$$

where the parameters $\lambda_{ij}$ are proportional to the force magnitude and the sum is computed for all bonds of the $i^{th}$ bead.

In the SHAKE algorithm, $\lambda_{ij}$ are treated as Lagrangian multipliers [29] which are determined by the requirement that the corresponding length constraints in Eq. (2) are satisfied exactly at each time step. This leads to the following two-step algorithm:

1. Calculate $\boldsymbol{r}_i'(t+\Delta t)$ - the unconstrained positions of beads neglecting the effect of constraint forces.
2. Determine the correction vectors $\delta \boldsymbol{r}_i(t+\Delta t)$ and construct the constrained particle positions $\boldsymbol{r}_i(t+\Delta t) = \boldsymbol{r}_i'(t+\Delta t) + \delta \boldsymbol{r}_i(t+\Delta t)$

The correction vectors of the second step are given by [29]

$$\delta \boldsymbol{r}_i(t+\Delta t) = -\frac{\Delta t^2}{2m_i} \sum_{j=1\, j\neq i}^{N} \lambda_{ij} \left. \frac{\partial \sigma_{ij}}{\partial \boldsymbol{r}_i} \right|_t, \qquad [6]$$

where the derivative of the bond deviation is expanded as

$$\left. \frac{\partial \sigma_{ij}}{\partial \boldsymbol{r}_i} \right|_t = \frac{\partial \left( \left[ \boldsymbol{r}_i(t) - \boldsymbol{r}_j(t) \right]^2 - d_{ij}^2 \right)}{\partial \boldsymbol{r}_{ij}(t)}. \qquad [7]$$

The direction of the constraint forces in the SHAKE algorithm is determined by the orientation of the bonds taken from the previous time step. After substituting the constrained particle coordinates evaluated at $t+\Delta t$ and rearranging the terms, Eq. (2)

yileds

$$2(\boldsymbol{r}_i'-\boldsymbol{r}_j')\cdot\left(\delta\boldsymbol{r}_i-\delta\boldsymbol{r}_j\right)+\left(\delta\boldsymbol{r}_i-\delta\boldsymbol{r}_j\right)^2=d_{ij}^2-\left(\boldsymbol{r}_i-\boldsymbol{r}_j\right)^2. \quad [8]$$

Eqs. (6) and (8) result in a system of $\mathrm{N}-1$ quadratic equations for $\lambda_{ij}$ values. In order to avoid direct numerical solution of the resulting system, the SHAKE algorithm employs an iterative solution of the following linearised system on each time step

$$2(\boldsymbol{r}_i'-\boldsymbol{r}_j')\cdot\left(\delta\boldsymbol{r}_i^{(l)}-\delta\boldsymbol{r}_j^{(l)}\right)=d_{ij}^2-\left(\boldsymbol{r}_i-\boldsymbol{r}_j\right)^2-\left(\delta\boldsymbol{r}_i^{(l-1)}-\delta\boldsymbol{r}_j^{(l-1)}\right)^2, \quad [9]$$

where the index $(l-1)$ refers to the corrections vectors calculated using $\lambda_{ij}^{(l-1)}$ values obtained at the previous iteration step, and the index $(l)$ refers to the correction vectors at the current iteration step. The solution is initialised using $\lambda_{ij}^{(0)}=0$. Once the iterations have converged, the resulting $\lambda_{ij}$ values are used to construct the final correction vectors (Eq. (6)).

## 3 FALCO algorithm

Although the SHAKE algorithm is widely used in molecular dynamics simulations, the algorithm has limited convergence range and may even fail to converge when the initial guess is far from the final corrected particle position. While this problem usually does not manifest in molecular dynamics simulations, it has been observed in meta-model simulations previously [28]. The complications which arise from the limited convergence

range, namely the restrictions on the time step of unsteady coupled simulations prompted the development of the FALCO algorithm presented in this section.

In the FALCO algorithm the particle positions are iteratively determined using the mass-weighted deviation of the current bond length from the prescribed one as follows

$$\delta \boldsymbol{r}_i(t+\Delta t)^{(l)} = -\frac{\varepsilon}{2m_i}\left(\min\left(m_{i-1}, m_i\right) DL_{i-1,i}^{(l-1)} \boldsymbol{n}_{i-1,i}^{(l-1)} + \min\left(m_i, m_{i+1}\right) DL_{i,i+1}^{(l-1)} \boldsymbol{n}_{i,i+1}^{(l-1)}\right), \qquad [10]$$

where $DL_{i-1,i}^{(l-1)}$ denotes the elongation of the bond between particles $i$ and $i+1$ at $(l-1)$ iteration given by

$$DL_{i-1,i}^{(l-1)} = \sqrt{\left[\left(\boldsymbol{r}_{i-1}' + \delta \boldsymbol{r}_{i-1}^{(l-1)}\right) - \left(\boldsymbol{r}_i' + \delta \boldsymbol{r}_i^{(l-1)}\right)\right]^2} - d_{i-1,i}, \qquad [11]$$

where $\boldsymbol{n}_{i-1,i}^{(l-1)}$ is the unit vector along the direction of the bond between particles $i$ and $i-1$ at $(l-1)$ iteration; and $\varepsilon \in [0,1]$ is an under-relaxation factor.

The relation of the proposed corrector to the SHAKE algorithm can be established by considering an analogy with Hooke's law [30] with the spring constant proportional to the bond length elongation. The force then is proportional to the bond elongation with the coefficient of proportionality $\lambda_{i,j}$ (Eq. (5)). Let us consider the normalised bond elongation $DLN_{i,j}$ given by

$$DLN_{i,j} = \frac{\sqrt{\left[\left(\boldsymbol{r}_i' + \delta \boldsymbol{r}_i^{(l-1)}\right) - \left(\boldsymbol{r}_j' + \delta \boldsymbol{r}_j^{(l-1)}\right)\right]^2} - d_{i,j}}{\sqrt{\left[\left(\boldsymbol{r}_i' + \delta \boldsymbol{r}_i^{(l-1)}\right) - \left(\boldsymbol{r}_j' + \delta \boldsymbol{r}_j^{(l-1)}\right)\right]^2}}. \qquad [12]$$

Then one can define $\lambda_{i,j}$ to be proportional to the normalised bond elongation

$$\lambda_{i,j}^{(l)} = \frac{\varepsilon}{2} \cdot \min_{i \in bond} \left| \frac{m_i}{\Delta t^2} \right| \cdot DLN_{i,j}. \qquad [13]$$

Using this definition in the SHAKE correction formula (Eq. (6)) and taking into account the bond constraints updated on each iteration

$$\left. \frac{\partial \sigma_{i,j}}{\partial \boldsymbol{r}_i} \right|_{t+\Delta t}^{(l)} = \frac{\partial \left( \left[ \left( \boldsymbol{r}_i' + \delta \boldsymbol{r}_i^{(l-1)} \right) - \left( \boldsymbol{r}_j' + \delta \boldsymbol{r}_j^{(l-1)} \right) \right]^2 - d_{i,j}^2 \right)}{\partial \left( \boldsymbol{r}_i' + \delta \boldsymbol{r}_i^{(l-1)} \right)}, \qquad [14]$$

the FALCO update formula given by Eq. (10) is obtained. In the FALCO algorithm the particle positions are updated inside the iteration loop, which leads to the direction vectors being updated at each iteration, whereas the SHAKE algorithm relies on the direction vectors obtained at the previous time-step.. Thus, the FALCO algorithm is less sensitive to angular displacements of the bonds. The proposed formulation does not involve numerical solution of second-order equations, thereby reducing storage requirements and increasing computational efficiency.

## 4 Test cases

In order to validate the FALCO algorithm and demonstrate its stability, two test cases have been considered: (i) the rotating dumbell problem [31] where an analytic solution is

readily available. In this case, the mechanical model moves in a fluid at rest; (ii) a ssDNA motion in a lid-driven micro-cavity flow. For the second test case, the results obtained with SHAKE and FALCO algorithms are compared.

### 4.1 Rotating dumbbell

The theorem of zero net hydrodynamic force [31] states that the rotation of a non-skew body about any axis passing through its centre of hydrodynamic torque $C$ results in zero net hydrodynamic force. The centre of hydrodynamic torque is defined as a point where the torque $\boldsymbol{T}_C$ vanishes in case of placing a fixed dumbbell into a homogeneous, uniform velocity field.

Consider an asymmetric dumbbell in a solid-like vortex with the angular velocity $\boldsymbol{\omega}$ about an axis passing through $C$ (Figure 2). Beads 1 and 2 are moving through the surrounding fluid with velocities $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ respectively. Due to the motion, the beads experience Stokes drag forces given by

$$\boldsymbol{F}_i = -6\pi\eta\, b_i \boldsymbol{u}_i, \qquad i = 1,2 \qquad [15]$$

where $\eta$ is the dynamic viscosity and $b_i$ is the bead radius.

The velocities are given by

$$\boldsymbol{u}_i = \boldsymbol{\omega} \times \boldsymbol{r}_{Ci}, \qquad i = 1,2 \qquad [16]$$

which leads to the following expression for the total force

$$\boldsymbol{F} = -6\pi\eta\,\boldsymbol{\omega}\times\left(b_1\boldsymbol{r}_{C1} + b_2\boldsymbol{r}_{C2}\right). \qquad [17]$$

The location of the centre of hydrodynamic torque $C$ is calculated from the following relation

$$\boldsymbol{r}_{C1} = -\frac{b_2}{b_1}\boldsymbol{r}_{C2}. \qquad [18]$$

Substituting this relation into the formula for the total force given by the Eq. (17) yields $\boldsymbol{F} = \boldsymbol{0}$, in accordance with the general theorem. A dumbbell with zero initial velocity at $t = 0$ should start rotating around its centre of hydrodynamic torque $C$ and accelerate until its angular velocity reaches that of the flow. The steady state is reached when the angular velocity of the dumbbell matches that of the flow. This problem can be used in order to validate the implementation of the rigid bond constraint. In the absence of the bond constraint, drag forces should result in spiral particle pathlines. Furthermore, an inconsistent or biased particle corrector algorithm would result in non-circular particle trajectories.

To verify the FALCO algorithm, the accelerating dumbbell problem has been computed for the beads with the radius ratio $b_2 / b_1 = 2.0$ and constant density corresponding to bead mass ratio of $m_2 / m_1 = 8.0$. The angular velocity of the clockwise rotating vortex was set to $1.0$. The bond length was equal to $0.2$. The orbits of the beads over $627$ time steps formed two concentric circles (Figure 3). The integration error lead to the

orbit displacement of $1\%$ after a complete circle for the time step of $0.01$. Figure 4 shows the deviation of the calculated bond length from the exact value normalised by the prescribed bond length ($DLN_{1,2}$).

4.2 ssDNA in lid-driven cavity

The flow in a square lid-driven micro-cavity was investigated as an example of a more practical scenario of a macromolecule transport. In this case, coupled solution for the flow field and the motion of a small ssDNA segment has been obtained. Using the bead-rod model, the ssDNA segment was represented by 3 beads connected with rigid and freely rotating rods. Simulations were performed for two different values of the time step with both SHAKE and FALCO algorithms. The flow was computed using the in-house HIRECOM Navier-Stokes finite-volume solver based on artificial compressibility pressure-velocity coupling approach [34] and a characteristics-based high-resolution scheme [35]. The solution was integrated in pseudo-time using the 4th-order Runge-Kutta method and the real-time integration was performed using a 2nd-order backwards differencing (e.g. [36]).

At time $t = 0$ the ssDNA strand was injected into a developed micro-cavity flow with the Reynolds number of 10 based on the cavity depth $L$ and the lid velocity $U$. The ssDNA was aligned vertically, downwards from the starting point given by $x_0 = 0.5$, $y_0 = 0.2$.

Since the primary aim of the test case was to evaluate the convergence range of the algorithm, the force exerted on the fluid flow by the ssDNA was neglected in order to

simplify the problem. Estimates performed for the conditions investigated in this paper indicate that this external force is 4 to 7 orders of magnitude smaller than the convective and viscous terms. The stochastic Brownian force acting on the beads of the mechanical model [28] was also neglected in order to enable direct deterministic comparison of pathlines obtained with different particle correctors. Considering motion of the investigated Brownian particles in fluid at room temperature, stochastic forces are 6 orders of magnitude smaller compared to viscous drag forces. Under these assumptions the equations determining the motion of the beads without constraint forces reduce to

$$m_i \frac{d^2 \boldsymbol{r}_i}{dt^2} = 6\pi\eta b_i \left(\boldsymbol{v}(\boldsymbol{r}_i) - \boldsymbol{u}_i\right) \qquad i = 1..3 \qquad [19]$$

The corresponding length constraints are given by

$$\left|\boldsymbol{r}_i - \boldsymbol{r}_{i+1}\right| = d_{i,i+1}. \qquad [20]$$

The density of the beads, bond lengths and bead radii were obtained from the experimental data published by Tothova et al. [37], which yields the following bead parameters normalised by the fluid density $\rho$ at room temperature and the cavity depth of $L = 1.0\mu m$

$$\rho_{1-3} = 1.22 \cdot 10^{-8}, \qquad b_{1-3} = 3.9 \cdot 10^{-3}, \qquad d_{12} = d_{23} = 8.3 \cdot 10^{-3} \qquad [21]$$

The flow was computed on a uniform orthogonal grid comprising $64 \times 64$ computational cells. The force acting on the ssDNA beads was obtained based on the volume-averaged flow velocity in the corresponding flow cell. No-slip boundary conditions were imposed on the cavity walls. The computations were performed with $\Delta t = 0.05$ and $\Delta t = 0.01$.

Initial simulations were performed using the small time step $\Delta t = 0.01$ for $10000$ time steps. Figure 5 shows particle positions obtained with the FALCO algorithm. Positions are displayed with $\delta t = 0.2$ sampling frequency, up to the time step $1200$. Figure 6 shows the average difference between the bead positions obtained with both algorithms defined as:

$$D = \frac{1}{3}\sum_{i=1}^{3}\left|\boldsymbol{r}_{SHAKE}^{i} - \boldsymbol{r}_{FALCO}^{i}\right|. \qquad [22]$$

Here, the data is shown prior to the first collision with the solid wall. After the collision, the ssDNA starts to spin and the average deviation increases. This can be expected since the update of the bond directions in the correction process of the FALCO algorithm results in better resolution of the rotational motion. However, prior to the wall collision and the fast spinning of the particle initiates, the difference between the results obtained with SHAKE and FALCO algorithms was negligible.

When increasing the time step to $\Delta t = 0.05$, the simulation using the SHAKE algorithm diverged after $145$ steps, whereas the simulation using the FALCO algorithm successfully converged. The results clearly demonstrate that the time step magnitude has an impact on the stability of the SHAKE algorithm. An explanation can be proposed

based on the fact that the SHAKE algorithm utilises previous time step bond directions, which leads to high sensitivity to the angular displacement of the bonds. The combination of angular deflections and large deviations from the prescribed bond length leads to the divergence of the particle corrector. The validity of this hypothesis can be demonstrated by analysing the evolution of the initial guess for the particle corrector. The quality of the initial guess can be quantified using the average deviation of the unconstrained bond length from the prescribed bond length given by

$$\Delta d = \frac{1}{2}\left(\left|d_{12}^{unc} - d\right| - \left|d_{23}^{unc} - d\right|\right) \qquad [23]$$

and the average angular displacement given by

$$\Delta\alpha = \frac{1}{2}(\alpha_{12} + \alpha_{23}), \qquad [24]$$

where $\alpha_{12}$ and $\alpha_{23}$ denote the angles between the bond orientation at time level $t$ and $t + \Delta t$ for bonds 12 and 23 respectively.

Figures 7 and 8 show the evolution of $\Delta d$ and $\Delta\alpha$ for the simulation conducted with the SHAKE algorithm using time step $\Delta t = 0.05$. As the simulation progresses, the initial guess gets worse for both measures. Both measures increase when the SHAKE simulation diverges with the average angular deviation equal to $2.83^0$ and the average bond length deviation equal to $0.99\%$. However, whereas for the average angular deviation the divergence point corresponds to a maximum, the same is not true for the average bond length deviation.

Figures 9 and 10 show the evolution of $\Delta d$ and $\Delta\alpha$ measures for the simulation conducted with the FALCO algorithm and the same time step $\Delta t = 0.05$. As can be seen from Figure 9, $\Delta d$ reaches the value of $1.22\%$ which is higher than the value of $\Delta d$ at the divergence point of the SHAKE algorithm. Furthermore, during the simulation conducted with the FALCO algorithm, the angular displacement $\Delta\alpha$ reaches the value of $16.3^0$ at time step $134$, which exceeds significantly the critical value observed at the divergence point of the SHAKE algorithm.

The results indicate that it is possible to obtain a converged solution with the FALCO algorithm even when the initial guess is much worse ($\Delta\alpha_{FALCO} = 16.3^0$) than that leading to the breakdown of the SHAKE algorithm ($\Delta\alpha_{SHAKE} = 2.83^0$).

## 5 Conclusions

A new particle corrector algorithm has been proposed based on purely geometric considerations. The final update formula of the proposed FALCO algorithm can be related to the SHAKE particle corrector of Ryckaert et al. [29] by an appropriate choice of Lagrangian multipliers. Although both SHAKE and FALCO algorithms are based on the method of constraint forces, numerical implementation of force approximations reveals significant differences. Contrary to SHAKE algorithm, the developed FALCO corrector approximates the constraint forces a-priori. This approach reduces the computational requirements and enhances overall stability of the new algorithm. The other beneficial feature of the proposed algorithm is the iterative update of the bond directions, which reduces sensitivity of the FALCO algorithm to angular displacements.

The FALCO algorithm has been validated using the analytic solution for the rotating dumbbell problem. Further numerical tests demonstrate that the proposed FALCO algorithm is more robust and capable of obtaining converged solution with relatively poor initial guess, which ultimately allows utilisation of larger time steps.

## 6 Acknowledgments

This work has been supported in part by the European Commission under the 6th Framework Program (Project: DINAMICS, NMP4-CT-2007-026804).

7 References

[1] Freemantle M. Microscale technology. *Chemical and Engineering News*, 1999, **22**, 27-36.

[2] Ehrfeld W. and Hessel V. and Lowe H. *Microreactors: New Technology for Modern Chemistry,* Wiley-VHC, Weiheim, 2000.

[3] Nguyen N. T. and Wu Z. Micromixers - a review. *Journal of Micromechanics and Microengineering*, 2005, **15**, R1-R16.

[4] Squires,T. M. and Quake,S. R. Microfluidics: Fluid physics at the nanoliter scale. *Reviews of Modern Physics*, 2005, **77**, 977-1026.

[5] Whitesides G. The origins and the future of microfluidics. *Nature*, 2006, **442**, 368-373.

[6] Stone H. A. and Stroock A. D. and Ajdari A. Engineering flows in small devices: Microfluidics toward a lab-on-a-chip. *Annual Review of Fluid Mechanics*, 2004, **36**, 381-411.

[7] Stroock, A. D. and Dertinger, S. K. W. and Ajdari, A. and Mezic, I. and Stone, H. A. and Whitesides, G. M. Chaotic mixer for microchannels. *Science*, 2002, **295**, 647-651.

[8] Ottino J. M., Wiggins S. Introduction: mixing in microfluidics. *Philosophical Transactions of Royal Society London A*, 2004, **362**, 923-935.

[9] Zhang, Y. and Barber, R. W. and Emerson, D. R. Part Separation in Microfluidic

Devices - SPLITT Fractionation and Microfluidics. *Current Analytical Chemistry*, 2005, **1**, 345-354.

[10] Gargiuli J. and Shapiro E. and Gulhane H. and Nair G. and Drikakis D. and Vadgama P. Microfluidic systems for in situ formation of nylon 6,6 membranes, *Journal of Membrane Science.*, 2006, **282**, 257-265.

[11] Pasas, S. A. and Lacher, N. A. and Davies M. I. and Lunte, S. M. Detection of homocysteine by conventional and microchip capillary electrophoresis/electrochemistry. *Electrophoresis*, 2002, **23**, 759-766.

[12] Fanguy, J. C. and Henry, C. S. The analysis of uric acid in urine using microchip capillary electrophoresis with electrochemical detection. *Electrophoresis*, 2002, **23**, 767–773.

[13] Shi, Y. N. and Simpson, P. C. and Scherer, J. R. and Wexler, D. and Skibola, C. and Smith, M. T. and Mathies, R. A. Radial Capillary Array Electrophoresis Microplate and Scanner for High-Performance Nucleic Acid Analysis. *Analytical Chemistry*, 1999, **71**, 5354-5361.

[14] Simpson, J. W. and Ruiz-Martinez, M. C. and Mulhern, G. T. and Berka, J. and Latimer, J. R. and Ball, J. A. and Rothberg J. M. and Went, G. T. A transmission imaging spectrograph and microfabricated channel system for DNA analysis. *Electrophoresis*, 2000, **21**, 135–149.

[15] Agarwal, U.S. and Dutta, A. and Mashelkar, R. A. Migration of macromolecules under flow: the physical origin and engineering implications. *Chemical Engineering Science*, 1994, **49**, 1693-1717.

[16] Blom, M. T. and Chmela, E. and Gardeniers, J. G. E. and Tijssen, R. and

Elwenspoek, M. and van den Berg, A. Design and fabrication of a hydrodynamic chromatography chip. *Sensors and Actuators B*, 2002, **82**, 111-116.

[17] Wang, P.-C. and Gao, J. and Lee, C. S. High-resolution chiral separation using microfluidics-based membrane chromatography. *Journal of Chromatography A*, 2002, **942**, 115–122.

[18] Stein, D. and van der Heyden, F. H. J. and Koopmans, W. J. A. and Dekker, C. Pressure-driven transport of confined DNA polymers in fluididc channels. *PNAS*, 2006, **103**, 15853-15858.

[19] Wong P. K. and Lee Y.-K. and Ho C.-M. Deformation of DNA molecules by hydrodynamic focusing. *Journal of Fluid Mechanics*, 2003, **497**, 55-65.

[20] Nonaka, A. and Gulati S. and Trebotich, D. and Miller, G. H. and Muller, S. J. and Liepmann, D. Computational Model with Experimental Validation for DNA Flow in Microchannels. *NSTI-Nanotechnology*, 2005, **3**, 712-715.

[21] Chung Y.-C. and Lin Y.-C. and Hsu Y.-L. and Chang W.-N. T. and Shiu M.-Z. The effect of velocity and extensional strain rate on enhancing DNA hybridization. *Journal of Micromechanics and Microengineering*, 2004, **14**, 1376–1383.

[22] Shapiro E. and Drikakis D. and Gargiuli J. and Vadgama P. Microfluidic Cell Optimization for Polymer Membrane Fabrication. *Proceedings of the 4th ASME International Conference on Nanochannels, Microchannels and Minichannels*, Limerick, USA, 2006, ICNMM2006-96221.

[23] Karniadakis G. and Beskok A. and Aluru N. *Microflows and Nanoflows: Fundamentals and Simulation*. Springer, New York, 2005

[24] Drikakis D. and Kalweit M. Coupling strategies for hybrid molecular-continuum

simulation methods, Proc. IMechE: *J. Mechanical Engineering Science*, 2008, **222**, Part C, 797-806

[25] Gad-el-Hak M. Liquids: The holy grail of microfluidic modelling. *Physics of Fluids*, 2005, **17**, 100612.

[26] Cussler E. L. *Diffusion: Mass Transfer in Fluid Systems*. Cambridge University Press, Cambridge, 1997.

[27] Doi M. and Edwards S. F. *The theory of polymer dynamics*. Oxford: Clarendon, Oxford, 1986.

[28] Trebotich D. and Miller G. H. and Colella P. and Graves D. T. and Martin D. F. and Schwartz P. O. A tightly coupled particle-fluid model for DNA-laden flows in complex microscale geometries. *Proceedings of Computational Fluid and Solid Mechanics 2005*, M. I. T., USA, 2005 UCRL-CONF-208132

[29] Ryckaert J. P. and Ciccotti G. and Berendsen H. J. C. Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. *Journal of Computational Physics*, 1977, **23**, 327-341

[30] Goldstein H. *Classical mechanics*. Addison-Wesley, 1959.

[31] Happel, J. and Brenner, H. *Low Reynolds number hydrodynamics*. Hague : Martinus Nijhoff Publishers, Hague, 1983.

[32] Smith S. B. and Cui Y. and Bustamante C. Overstretching B-DNA: the elastic response of individual double-stranded and single-stranded DNA molecules. *Science*, 1996, **271**, 795-799.

[33] Bustamante C. and Smith S. B. and Liphard J. and Smith D. Single-molecule studies of DNA mechanics. *Current opinion in structural biology*, 2000, **10**, 279-285

[34] Chorin A. J. Numerical Solution of the Navier-Stokes Equations. *Math. Comp*., 1968, **22**, 745-762.

[35] Shapiro E. and Drikakis D. Non-Conservative and Conservative Formulations of Characteristics-Based Numerical Reconstructions for Incompressible Flows. *International Journal of Numerical Methods for Engineering*, 2006, **66(9)**, 1466-1482.

[36] Drikakis D. and Rider W. *High-Resolution Methods for Incompressible and Low-Speed Flows*. Springer, 2004.

[37] Tothova J., Brutovsky B., Lisy V. Addendum to “Monomer motion in single- and double-stranded DNA coils”. arXiv:cond-mat/0701523v1 [cond-mat.soft], 2005.

8 Figures

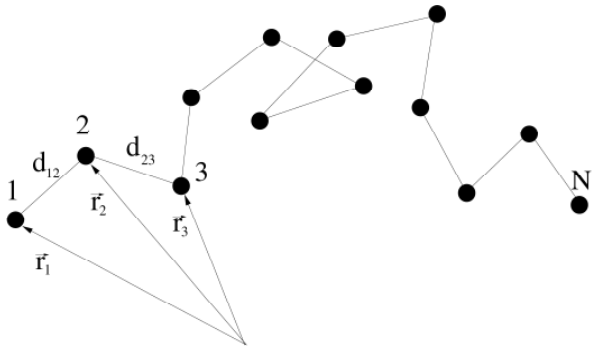


Figure 1

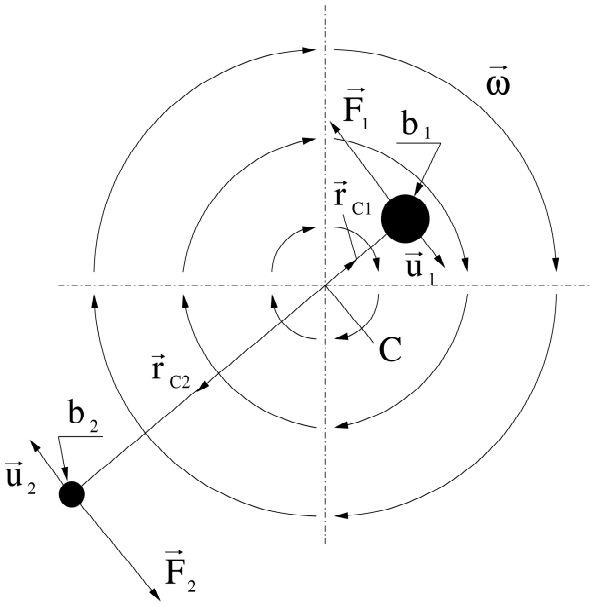


Figure 2

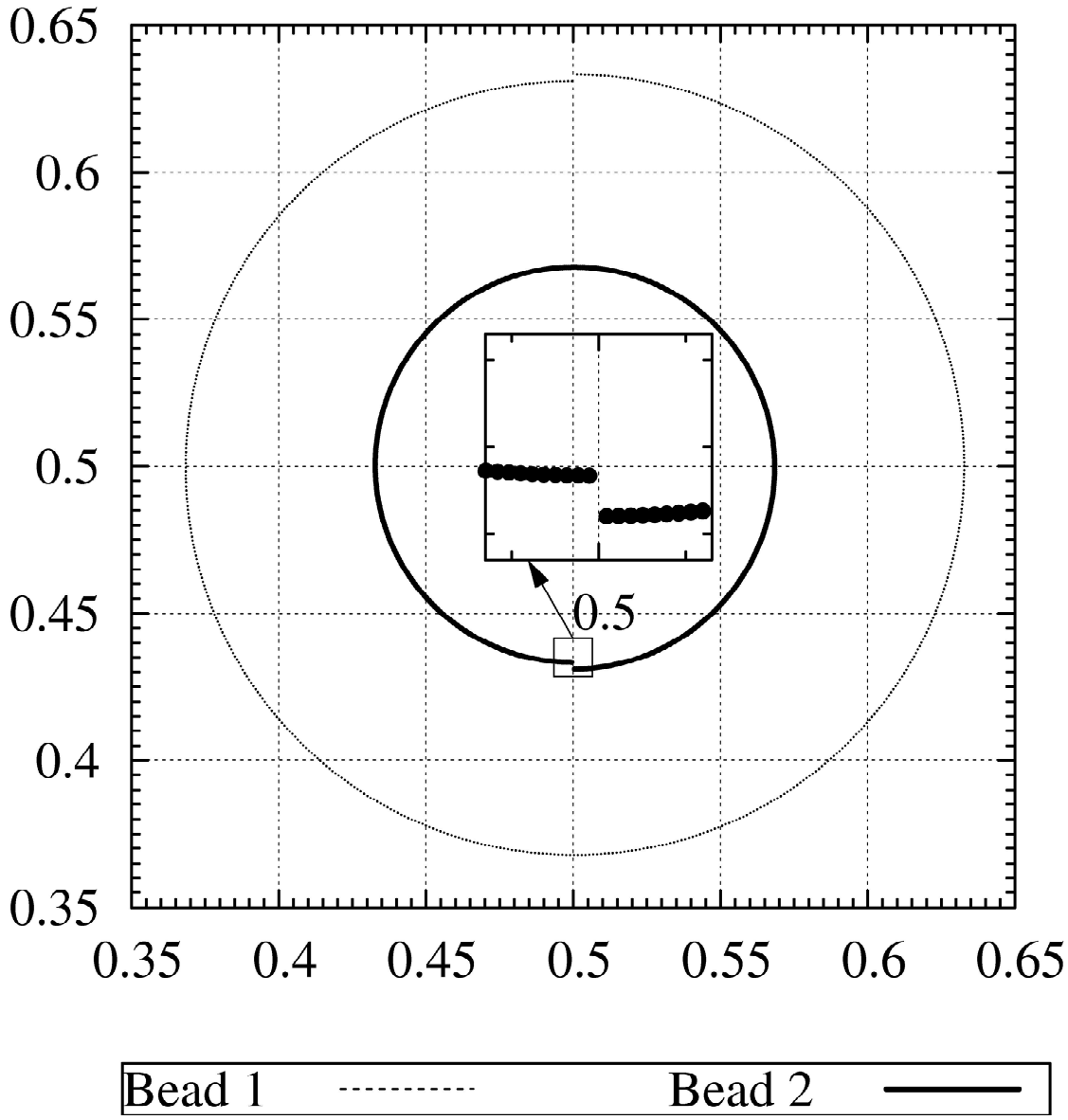


Figure 3

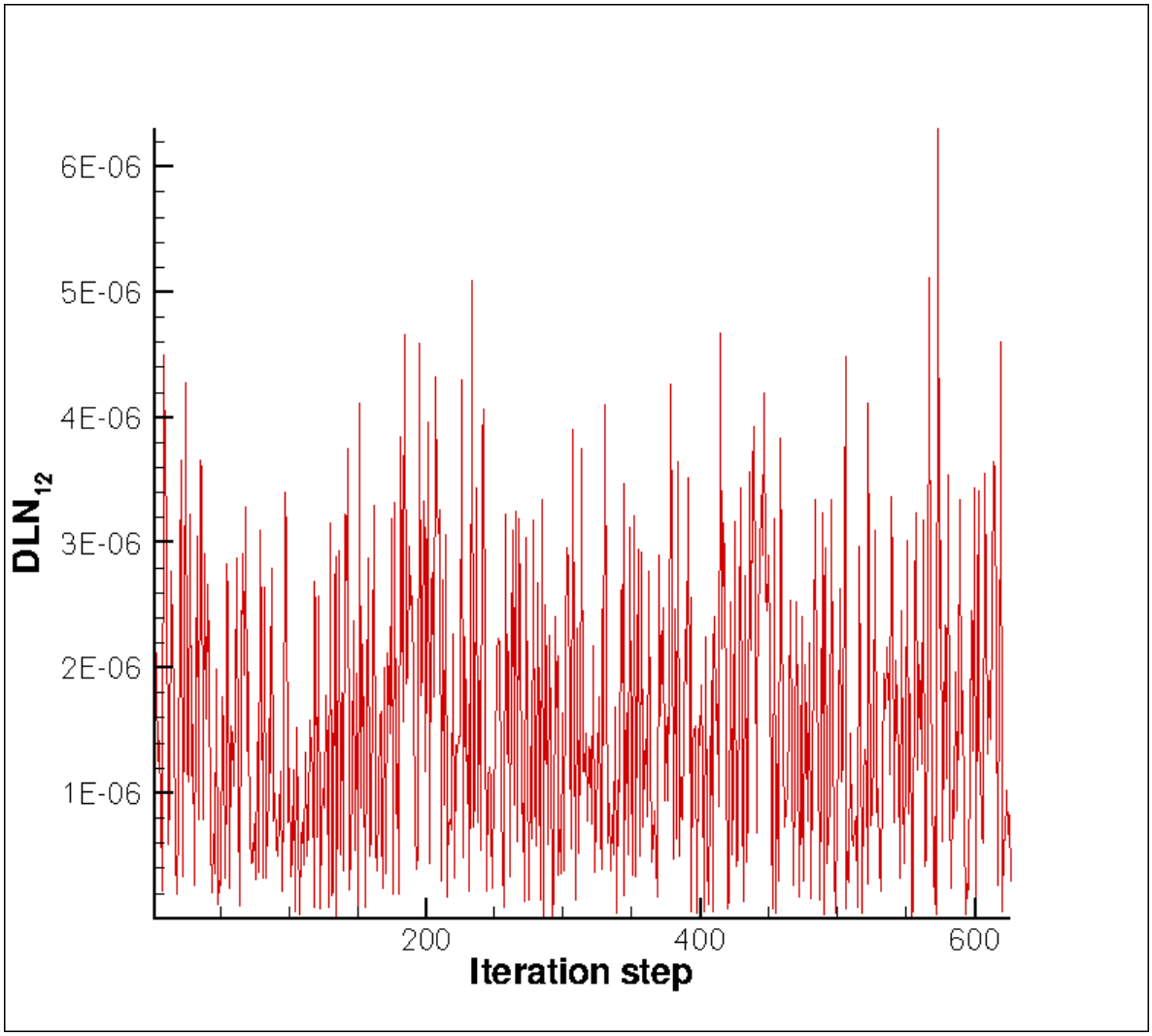
6E-06
5E-06
4E-06
3E-06
2E-06
1E-06
DLN12
200
400
600
Iteration step

Figure 4

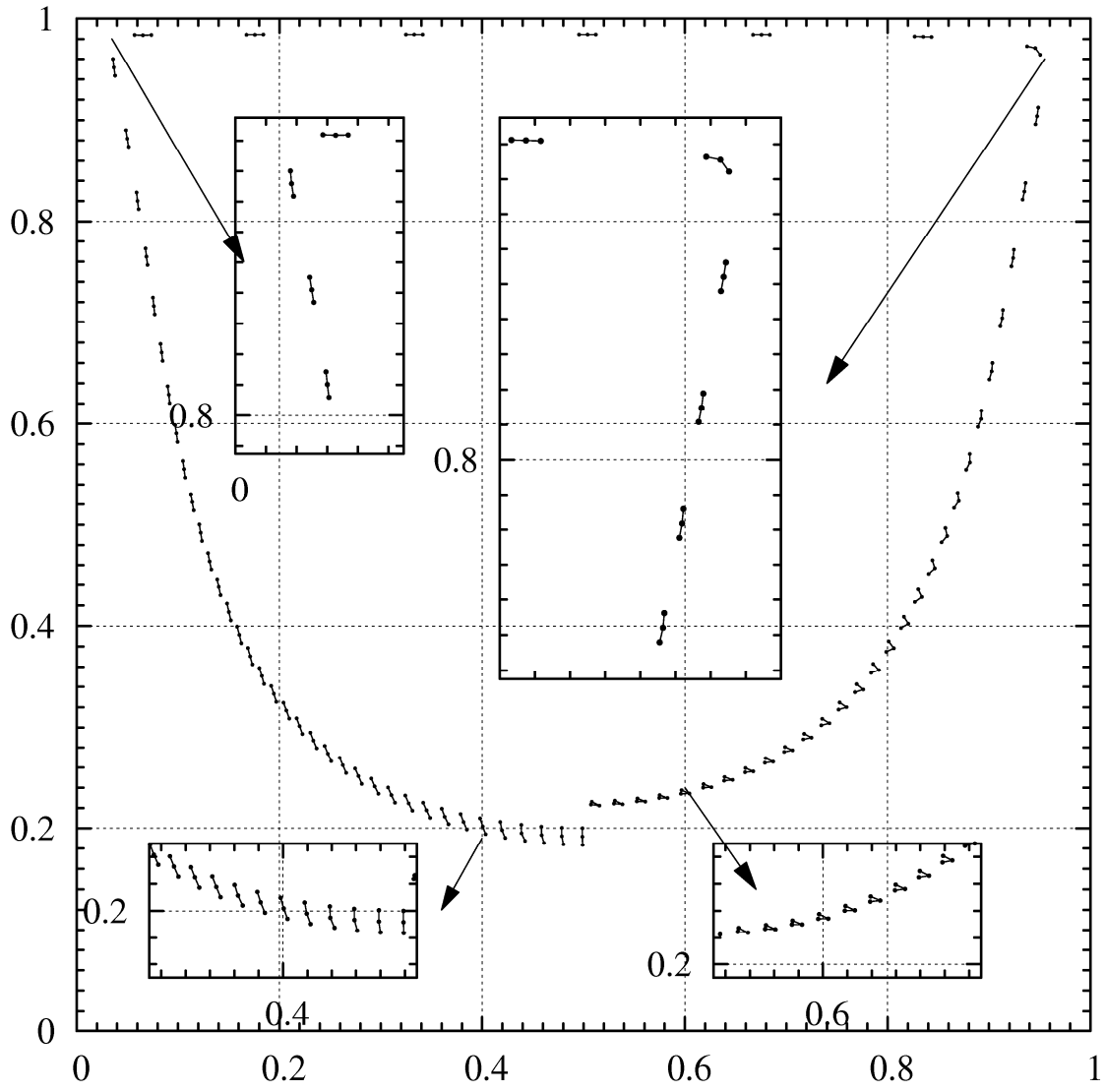


Figure 5

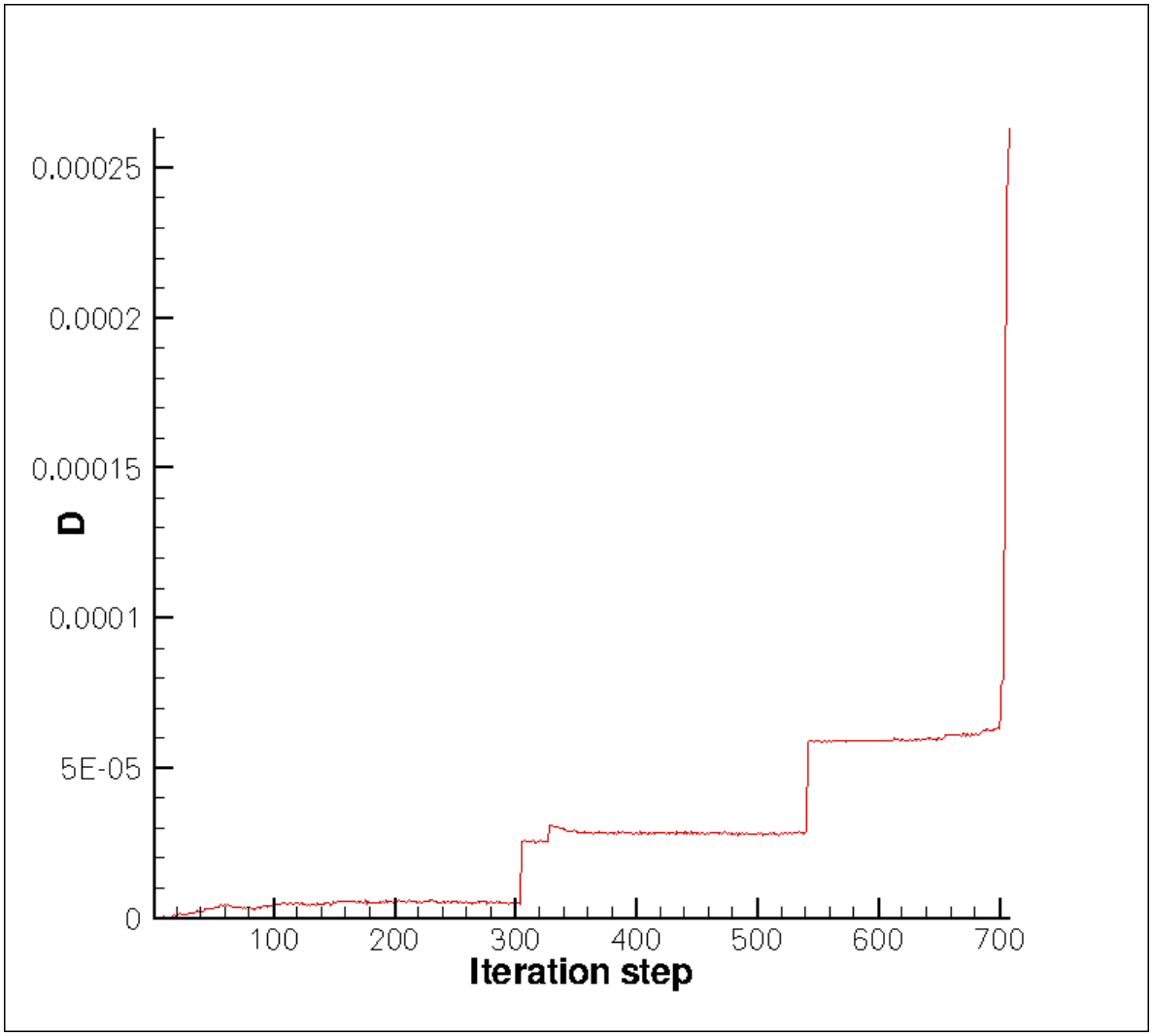
0.00025
0.0002
0.00015
D
0.0001
5E-05
0
100
200
300
400
500
600
700
Iteration step

Figure 6

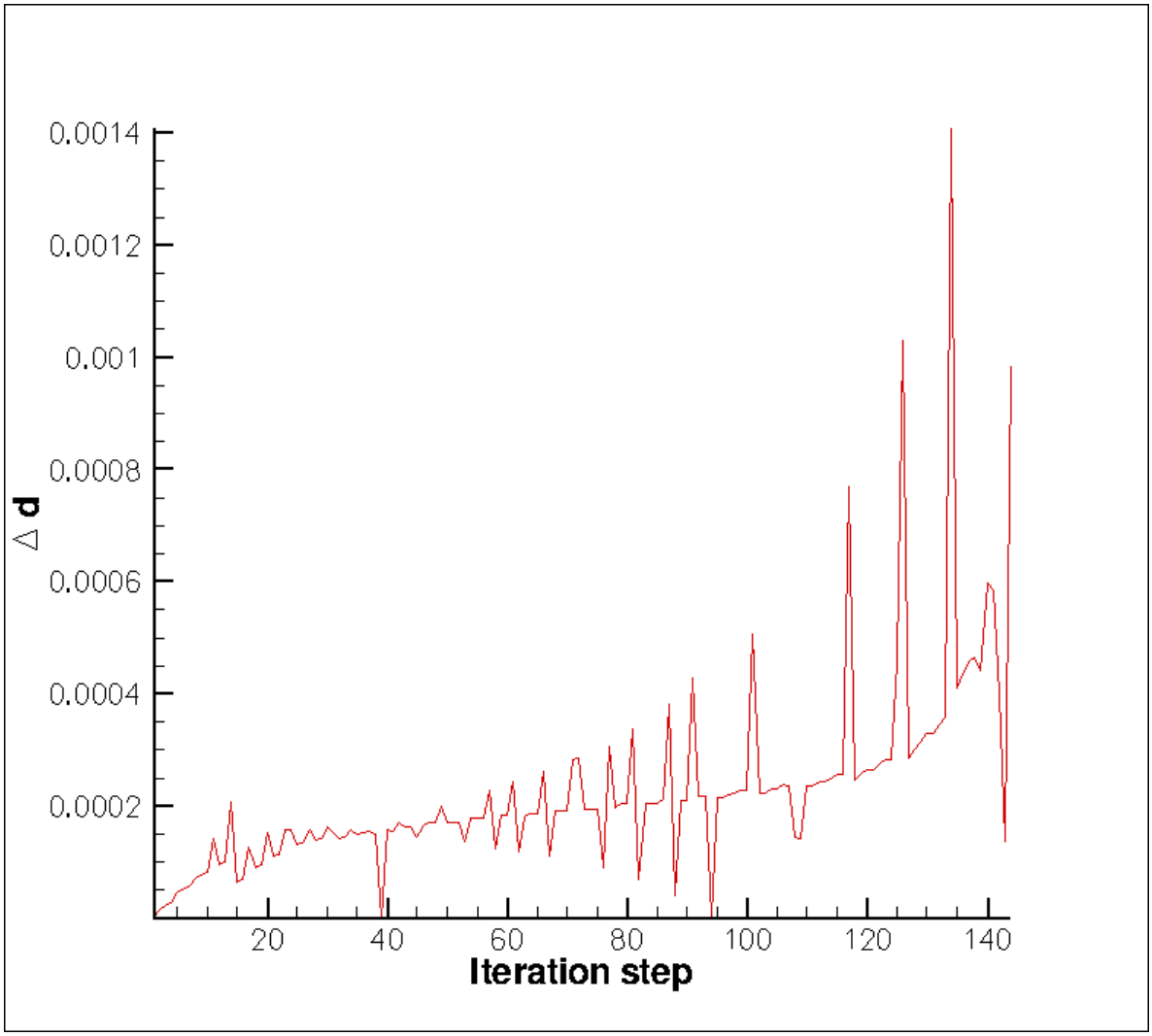
0.0014
0.0012
0.001
0.0008
0.0006
0.0004
0.0002
△ d
20
40
60
80
100
120
140
Iteration step

Figure 7

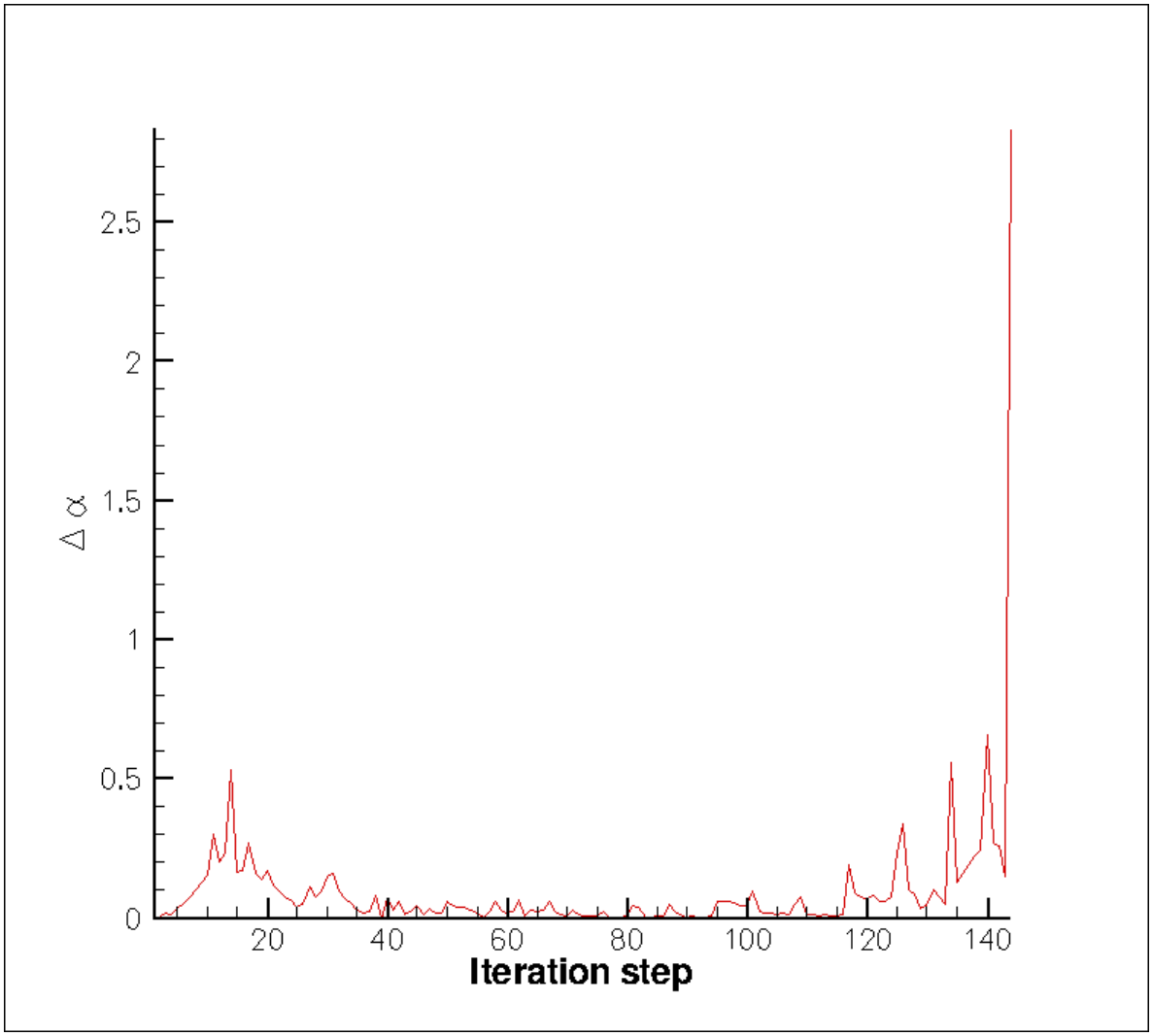


Figure 8

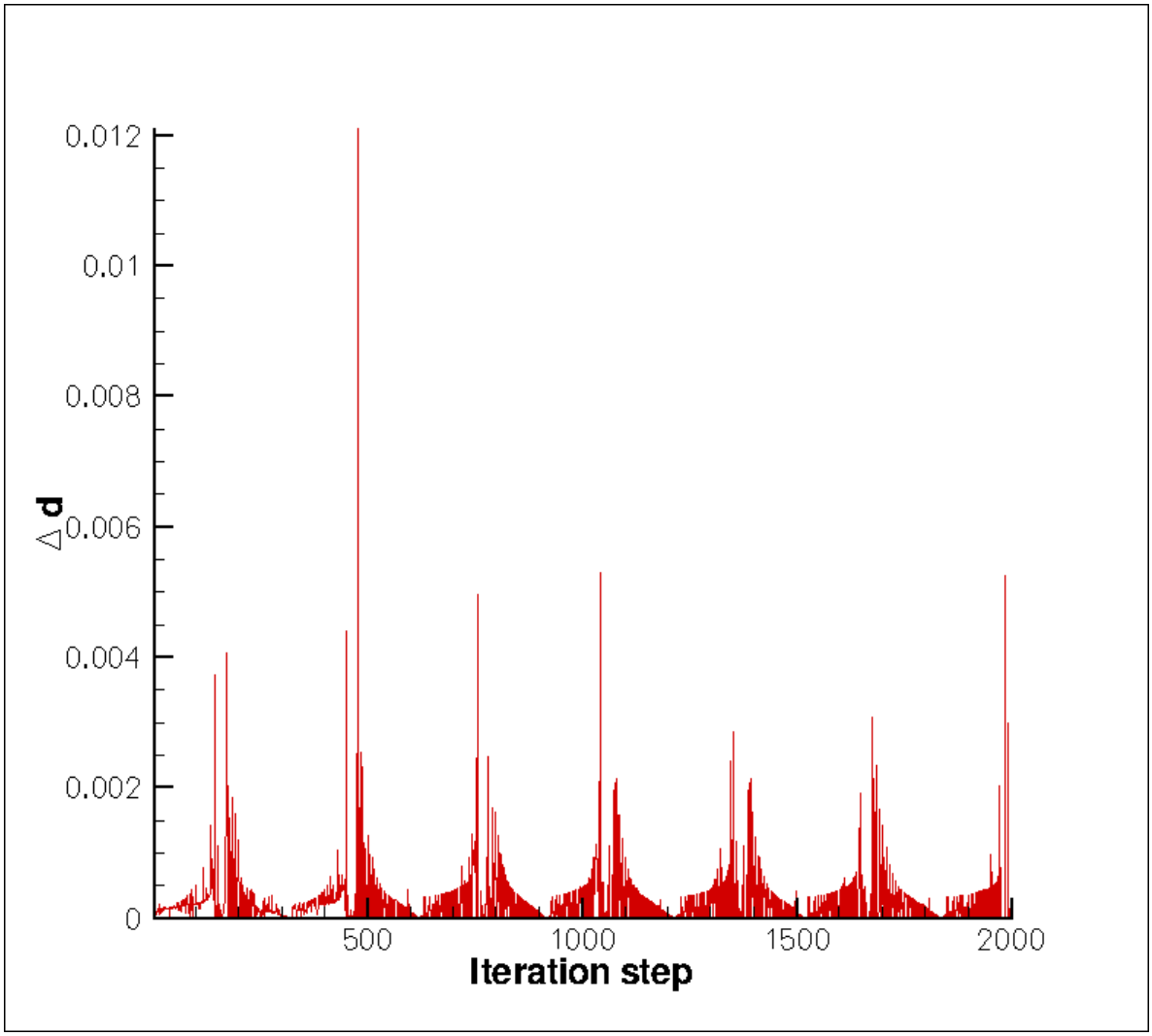
0.012
0.01
0.008
0.006
0.004
0.002
0
△d
500
1000
1500
2000
Iteration step

Figure 9

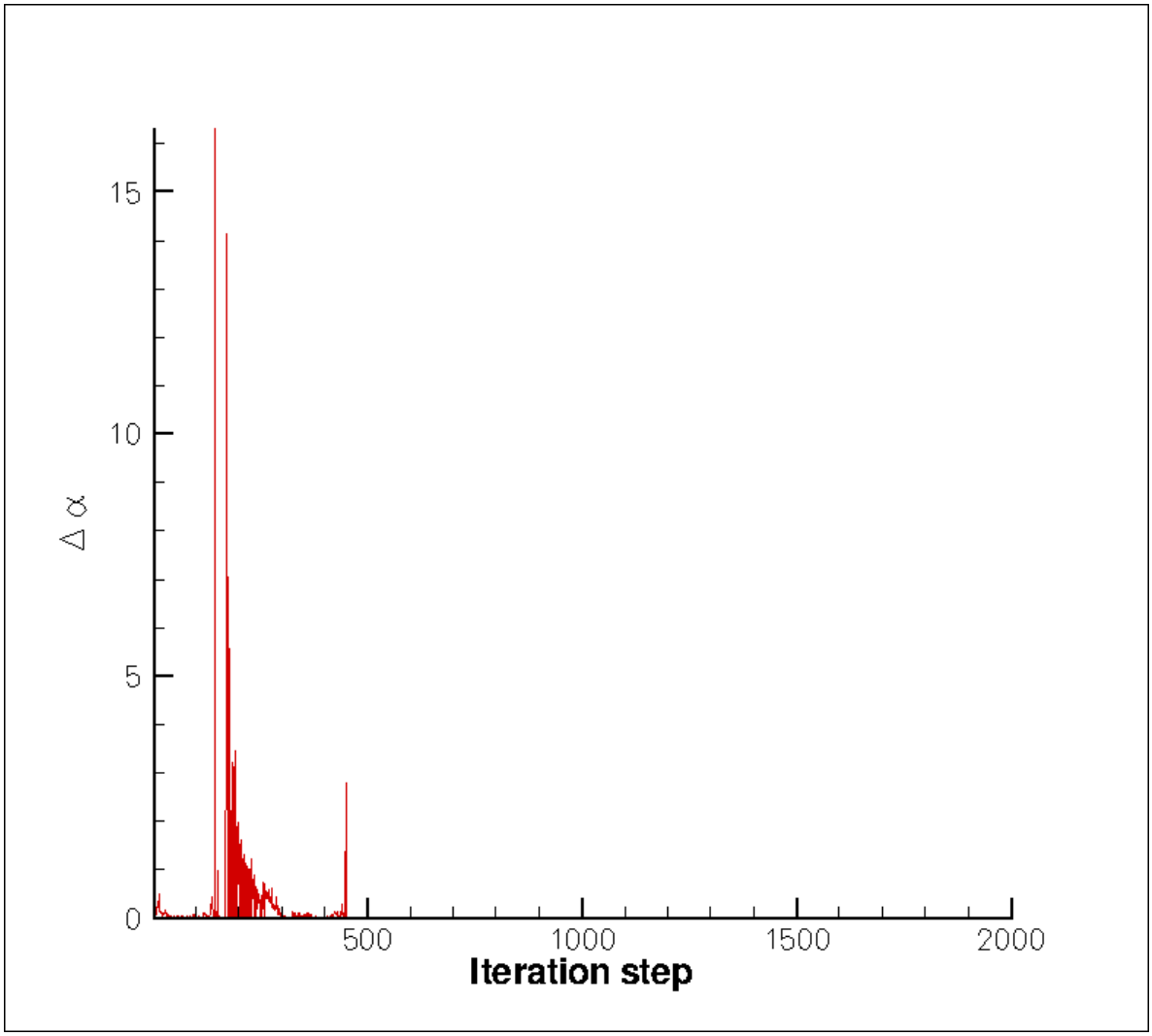


Figure 10

9 List of figure captions